VALID PARENTHESIS :
https://leetcode.com/problems/valid-parentheses/description/

```c
#define MAX 10
char stack[MAX];
int top = -1;
void push(char );
char pop(void );
char peek(void );

bool isValid(char* s) {
    int i = 0;

    while(s[i] != '\0') {
        if(s[i] == '(' || s[i] == '[' || s[i] == '{') {
            push(s[i]);
        }
        else if(s[i] == ')'){
            char ch = pop();
            if(ch != '(') return false;
        }
        else if(s[i] == ']'){
            char ch = pop();
            if(ch != '[') return false;
        }
        else if(s[i] == '}'){
            char ch = pop();
            if(ch != '{') return false;
        }
        i++;
    }
    if(top != -1) return false;
    return true;
}
void push(char ch) {
    if(top == MAX - 1) return;

    stack[++top] = ch;
}
char pop() {
    if(top == -1) return 'n';
    return stack[top--];
}
```

Testcase   Result

Accepted   Runtime: 0 ms

• Case 1   • Case 2   • Case 3

Input

s =
"()"

Output

false

Expected

false

Console ⌄